

## 1 . コンピュータの基本的な仕組み

コンピュータはプログラムの指示に従ってデータを入力して記憶し，記憶したデータについて四則演算などを行い，結果を出力する．また，コンピュータはこれらの一連の処理が正しい順序で行われるように制御を行う．コンピュータのこのような入力・記憶・演算・出力および制御をコンピュータの 5 大機能という．

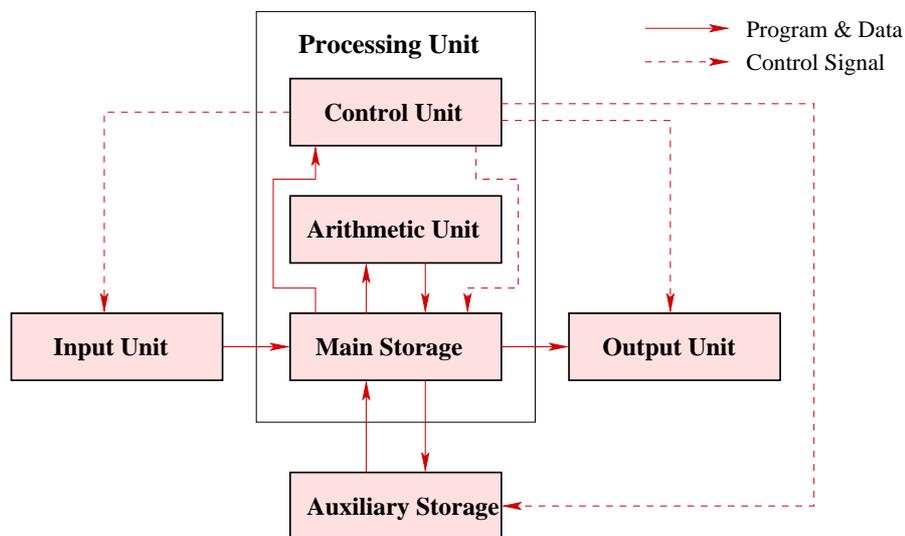


図 1: コンピュータの基本構成

- 入力装置 (input unit) : プログラム (program) やデータ (data) を入力する装置 .  
キーボード (keyboard) , OMR (Optical Mark Reader, 光学式マーク読取り装置) , OCR (Optical Character Reader , 光学式文字読取り装置) などがある .  
マウス (mouse) , ライトペン (light pen) , タッチスクリーン (touch screen) なども入力装置であるが , それらは指令を図形や絵 (アイコン) で選択させて処理させるグラフィカル・ユーザインタフェース (GUI: Graphical User Interface) において使用される (このようなコンピュータ操作の考え方をメタファ (隠喩) ともいう) . 位置のデータを入力する装置を総称して , ポインティング・デバイス (pointing device) などという .  
ほかに入力装置としては , バーコード (1次元) ・QRコード (2次元) リーダ , 音声入力装置 , イメージスキャナなどがある .
- 出力装置 (output unit) : 処理結果などを画面や紙面などに出力する装置 .  
表示装置 (display) , プリンタ (printer) などがある . コンピュータの表示装置 (モニタともいう) としては , ブラウン管 (CRT) よりも液晶ディスプレイ (LCD) が普通である .

- 記憶装置 (storage) : プログラムやデータを記憶する装置 .

記憶装置には , 主記憶装置 (main storage) と補助記憶装置 (auxiliary storage) がある . 主記憶装置は , 処理装置の内部にあってプログラムやデータを記憶し , 制御装置によって直接制御される . 補助記憶装置に比べて読み書きの速度は速いが , 記憶容量は小さい . 主記憶装置には主に半導体メモリ (IC, LSI, VLSI) が用いられる .

補助記憶装置は周辺装置の一つで , 主記憶装置の記憶容量を補う装置である . 主記憶装置に比べて読み書きの速度は遅いが , 記憶容量は大きい .

記憶の特性には「破壊読取り (destructive) / 非破壊読取り (nondestructive)」「揮発性 (volatile) / 不揮発性 (nonvolatile)」「消去可能 (erasable) / 永久 (permanent)」の区別がある .

補助記憶装置の磁気ディスク (HD : ハードディスク) やフロッピーディスクは非破壊 / 不揮発 / 永久メモリである .

一方 , 主記憶装置などの半導体メモリは非破壊であるが基本的には揮発性である (電源を切ると記憶した内容が消滅する) . 消去可能であり , どの場所 (address) にも (からも) 瞬時に書き込み・読み取り可能な RAM (Random Access Memory) とよばれるものである . もっとも , スタートアップ時などにおける基本プログラムは , 読み取り専用の不揮発 / 永久半導体メモリ ROM (Read Only Memory) に書き込んでおく .

USBフラッシュメモリ , メモリカード等は , この RAM と ROM の性質を兼ね備えた不揮発性半導体補助 (外部) 記憶装置である . 一方 , 光ディスクには CD (CD-ROM, CD-R, CD-RW), MO, DVD (DVD-ROM, DVD-R, DVD-RAM) がある .

- 演算装置 (arithmetic unit) : 四則演算や大小の比較判断などを行う装置 .

演算装置は , 主記憶装置から送られてくるデータについて , 加減乗除の四則演算や比較判断などを , 半導体による論理素子によって行う . 四則演算の基本は加算であり , 減算や乗算・除算は加算の原理を利用して行う .

- 制御装置 (control unit) : 入力・記憶・演算・出力装置を制御する装置 .

制御装置は , 主記憶装置に記憶されているプログラムの命令を一つずつ取り出して解読し , 各装置に指令を与える . 制御装置による命令の解読と実行 (execute) は , プログラムの順序に従って自動的に行う .

コンピュータの基本構成は図 1 に示すとおりであるが , このうち制御装置 , 演算装置 , 主記憶装置をまとめて処理装置 (processing unit) , とくに制御装置 , 演算装置を中央処理装置 (CPU : Central Processing Unit) という . 入出力装置や補助装置などは周辺装置 (peripheral equipment) とよばれる .

## 2 . 情報の表現

### 2.1 数の体系と基数, 2進数, 10進数

日常使用している数の表現は,  $0 \sim 9$  の 10 種類の記号を使い, 下位の桁から順に  $10^0, 10^1, 10^2, 10^3, \dots$  の重みがついている. このような表現を 10 進数 (decimal numeral(あるいは単に decimal number)) という. たとえば, 582 は

$$\begin{aligned} 582 &= 5 \times 100 + 8 \times 10 + 2 \times 1 \\ &= 5 \times 10^2 + 8 \times 10^1 + 2 \times 10^0 \end{aligned}$$

と表され, この 10 を基数 (radix) という.

- 2 進数 (binary numeral)

2 進数は基数が 2 で, 0 と 1 の 2 種類の記号 (数) を使用して表現する数値である. たとえば, 10 進数の 13 は 2 進数で

$$(1101)_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = (13)_{10}$$

と表される. 何進数の数であるかを区別するために, このように数を括弧で囲み右下に基数を書くことがある.

2 進数は, コンピュータ内部そしてデジタル伝送信号の情報表現の最小単位, ビット (bit, binary digit) による数値の表現である. 0 をたとえば低い電圧 1 を高い電圧と対応させることにより, 半導体回路によって比較判断や演算を行わせることができる.

- 8 進数 (octal numeral)

8 進数は基数が 8 で,  $0 \sim 7$  の 8 種類の記号 (数) を使用して表現する数値である. たとえば,

$$(572)_8 = 5 \times 8^2 + 7 \times 8^1 + 2 \times 8^0 = (378)_{10}$$

である.

- 16 進数 (hexadecimal numeral)

16 進数は基数が 16 で,  $0 \sim 9, A, B, C, D, E, F$  の 16 種類の記号 (数) を使用して表現する数値である. この  $A \sim F$  は, 10 進数の  $10 \sim 15$  に相当する数である. たとえば,

$$(3E6)_{16} = 3 \times 16^2 + 14 \times 16^1 + 6 \times 16^0 = (998)_{10}$$

と表される. 16 進数は, 8, 16, 32 ビットで処理されるコンピュータの数値を 4 ビットずつまとめて表現することができる. コンピュータ内部のデータコードそして機械語プログラムなどにおいて用いられる.

一般に,  $M$  進数の数  $N = a_n a_{n-1} \dots a_1 a_0$  は

$$(N)_M = a_n \cdot M^n + a_{n-1} \cdot M^{n-1} + \dots + a_1 \cdot M^1 + a_0 \cdot M^0$$

で表され、基数  $M$  での数値の表現法を  $M$  進法 (10 進法, 2 進法など) という。

整数の変換は上記の例のとおりであるが、小数点を含む数についても同様に変換することができる。たとえば、2 進数, 8 進数, 16 進数から 10 進数への変換例は次のとおりである：

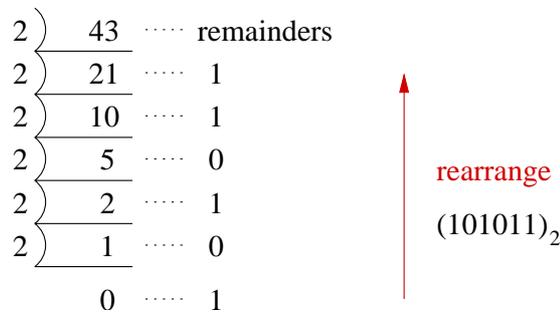
$$(10.011)_2 = 1 \times 2^1 + 0 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3} = (2.375)_{10},$$

$$(47.24)_8 = 4 \times 8^1 + 7 \times 8^0 + 2 \times 8^{-1} + 4 \times 8^{-2} = (39.3125)_{10},$$

$$(9.68)_{16} = 9 \times 16^0 + 6 \times 16^{-1} + 8 \times 16^{-2} = (9.40625)_{10}.$$

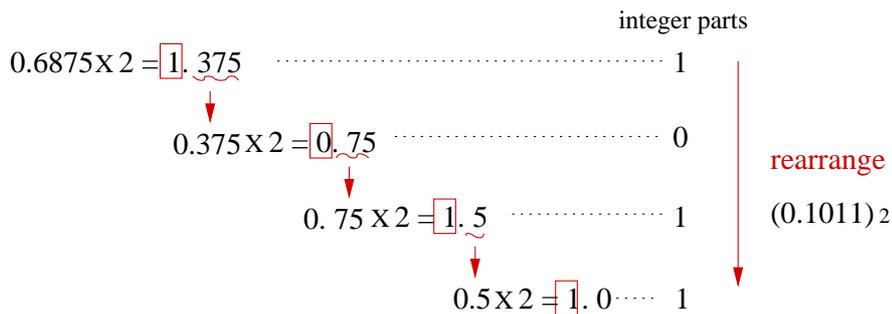
逆に、10 進数から 2 進数, 8 進数, 16 進数への変換は次のような手順による。以下、2 進数への変換のみ記す。

● 整数の変換



● 小数の変換

10 進数 0.6875 を 2 進数に変換



2.2 2 進数の計算

● 加算

10 進数による  $3 + 6 = 9$  の 2 進数による計算例を示す。

$$\begin{array}{r}
 0 \ 0 \ 1 \ 1 \\
 + \ 0 \ 1 \ 1 \ 0 \\
 \hline
 1 \ 0 \ 0 \ 1
 \end{array}$$

ここで， $1 + 0 = 1$  であるが， $1 + 1$  は桁上げ (carry) で， $(10)_2$  となることに注意．

- 減算

逆に，10 進数の  $9 - 6 = 3$  は

$$\begin{array}{r} 1\ 0\ 0\ 1 \\ -\ 0\ 1\ 1\ 0 \\ \hline 0\ 0\ 1\ 1 \end{array}$$

である．ここで， $1 - 0 = 1$  であるが， $0 - 1$  は 1 桁上からの借り (borrow) となることに注意．

- 補数の利用

補数 (complement) とは，ある数から別の数を引いて，その結果として得られる数である．2 進数の減算は，引く数の補数をつくり，その補数の加算によって求める．

2 進数の補数は次の手順により得られる．

- (1) 各桁において， $1 \rightarrow 0$ ， $0 \rightarrow 1$  と置き換える．これを 1 の補数という．
- (2) 1 の補数に 1 を加算する．これを 2 の補数という．

$(0110)_2 = (6)_{10}$  の補数を求める例を示そう．

$$\begin{array}{r} 0\ 1\ 1\ 0 \quad \dots (6)_{10} \\ \downarrow\ \downarrow\ \downarrow\ \downarrow \\ 1\ 0\ 0\ 1 \quad \dots 1\ \text{の補数} \\ +\ 0\ 0\ 0\ 1 \quad \dots 1\ \text{を加える} \\ \hline 1\ 0\ 1\ 0 \quad \dots 2\ \text{の補数} \end{array}$$

補数を使った減算  $9 - 6 = 3$  は次のようにして行う．

$$\begin{array}{r} 1\ 0\ 0\ 1 \quad \dots \text{引かれる数 } (9)_{10} \\ +\ 1\ 0\ 1\ 0 \quad \dots \text{引く数の } 2\ \text{の補数} \\ \hline (1)\ 0\ 0\ 1\ 1 \quad \dots \text{あふれた桁は無視する} \end{array}$$

## 2.3 文字 / 数字の表現

- 表現できる範囲

たとえば，10 進数 3 桁の場合， $0 \sim 999$ ， $10^3 = 1000$  種類 (通り) を表現することができる．一方，2 進数 4 桁では， $0 \sim 15$ ， $2^4 = 16$  種類，8 桁では， $0 \sim 255$ ， $2^8 = 256$  種類，さらに，16 進数では 4 桁の場合， $0 \sim 65535$ ， $16^4 = 65536$  種類を表わすことができる．一般に， $M$  進数  $n$  桁では， $0 \sim M^n - 1$  まで， $M^n$  種類 (通り) の表現ができる．

- ビットとバイト

ビット (bit, binary digit) は，コンピュータ内部そしてデジタル伝送信号の情報表現の最小単位であるが，通常は 2 進数の桁数をいうことが多い．文字 / 数字・記号など，とくにキーボード上の英数字は 8 ビットで表現されるため，8 ビットを 1 単位としてまとめてバイト (B, byte) という．

- 補助単位

記憶量など大きな単位を表す補助単位として、k(kilo-,  $10^3$ ), M(Mega-,  $10^6$ ), G(Giga-,  $10^9$ ) そして T(Tera-,  $10^{12}$ ) がある。一方、命令の実行速度がどれだけ (秒) といった小さな量を表す補助単位としては、m(milli-,  $10^{-3}$ ),  $\mu$ (micro-,  $10^{-6}$ ), n(nano-,  $10^{-9}$ ), p(pico-,  $10^{-12}$ ) などがある。

- コード (code)

数字, 英字, カナ, 記号などキーボード上の文字 (半角文字) (さらに漢字などの全角文字) はあらかじめ決められたコード (情報交換用符号) で表現される。このコードは国際的に共通に使用するため、ISO (International Organization for Standardization: 国際標準化機構) や JIS (Japanese Industrial Standards: 日本工業規格) などによって定められている。また、キーボード上の文字のコードとしては、汎用コンピュータ用の EBCDIC (Extended Binary Coded Decimal Interchange Code: 拡張 2 進化 10 進コード) そして PC 用の ASCII (American Standard Code for Information Interchange) が知られている。

ASCII では 7 ビットで 1 文字を表現し、これにパリティビット (誤り検査用ビット) を加えた 8 ビットのコードを ISO で制定している。JIS も正式には 7 bitJIS とよばれている。インターネット・メールなどの標準コードである。

日本の PC のコードはシフト JIS (Shifted JIS, SJIS) であるが、それは JIS コードの 8 ビット体系と重ならないよう、ずらして構成したからである。マイクロソフト社によって開発されたことから、MS-Kanji コードなどもよばれる。これらは、キーボード上の英数字の 7 ビット分については違いはない。

- パリティチェック (parity check: 奇偶検査)

データコードに誤りがなくどうか検査するビットを付加し、検査ビット (パリティビット) を含めて '1' の数を常に奇数または偶数になるようにする。常に奇数となるようにするデータコードを奇数パリティチェック、常に偶数となるようにするデータコードを偶数パリティチェックという。

- コード表

キーボード上の英数字のパリティのない場合の 8 ビットコードの例を以下に示す。

	b <sub>8</sub>	b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	16 進数
@	0	1	0	0	0	0	0	0	(40) <sub>16</sub>
A	0	1	0	0	0	0	0	1	(41) <sub>16</sub>
B	0	1	0	0	0	0	1	0	(42) <sub>16</sub>
C	0	1	0	0	0	0	1	1	(43) <sub>16</sub>
a	0	1	1	0	0	0	0	1	(61) <sub>16</sub>
b	0	1	1	0	0	0	1	0	(62) <sub>16</sub>
c	0	1	1	0	0	0	1	1	(63) <sub>16</sub>
\$	0	0	1	0	0	1	0	0	(24) <sub>16</sub>
%	0	0	1	0	0	1	0	1	(25) <sub>16</sub>
&	0	0	1	0	0	1	1	0	(26) <sub>16</sub>

## 2.4 数値の表現

- 数値データの種類

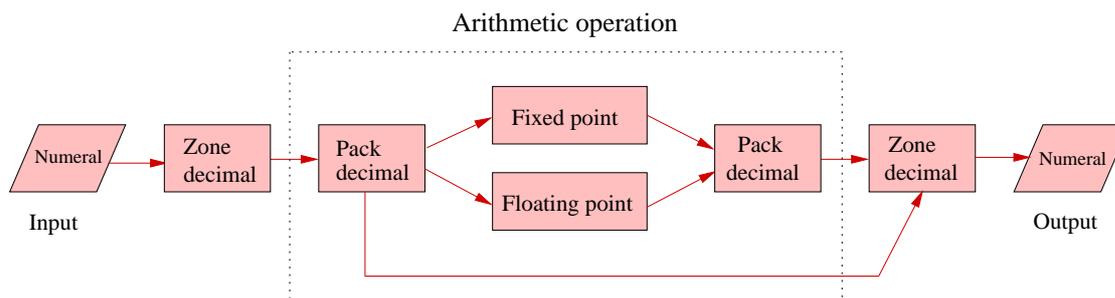
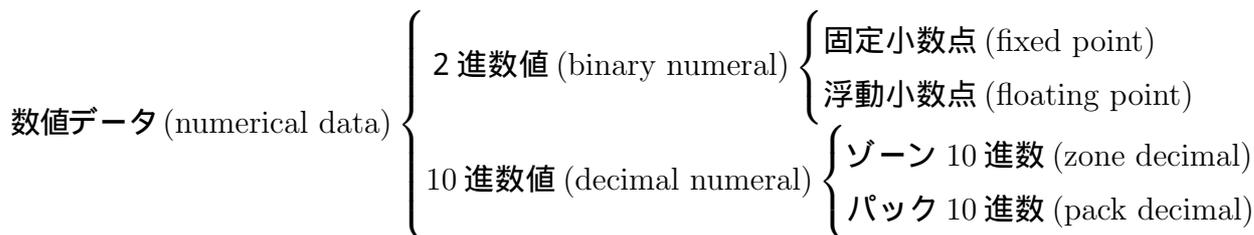


図 2: コンピュータにおける演算処理

- 固定小数点データ (fixed point data)

固定小数点データは、図 3 に示すように、最上位の桁で正または負の符号を表し (正は「0」、負は「1」)、残りの桁で数値を表す。負の数値は 2 の補数を使用される。

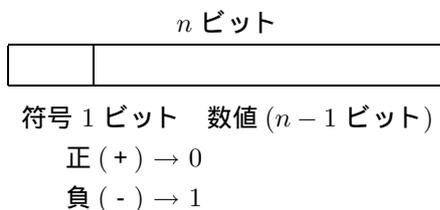


図 3: 固定小数点データの表現

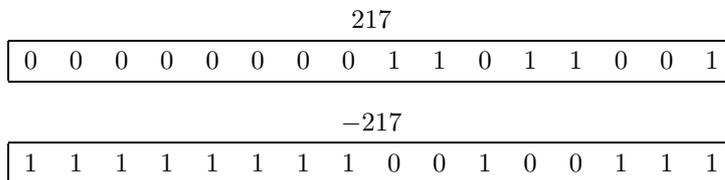


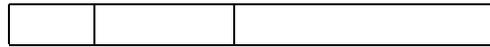
図 4: 217 と -217 の固定小数点データ (16 ビット) の表現

表現できる範囲 固定小数点では、2 進数  $n$  桁で表現できる数値の範囲は、 $-2^{n-1} \sim 2^{n-1} - 1$  である。16 ビットでは、 $-2^{15} = -32768 \sim 2^{15} - 1 = 32767$  である。

- 浮動小数点データ (floating point data)

正規化(normalization)

32ビットまたは64ビット(32ビット:単精度,64ビット:倍精度)



仮数部の符号 1ビット 指数部(7ビット) 仮数部(24ビットまたは56ビット)  
 正(+) → 0  
 負(-) → 1

図 5: 浮動小数点データの表現

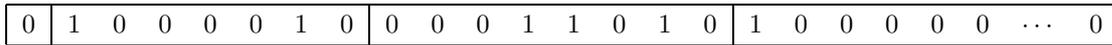


図 6: 26.5 を 32 ビットの浮動小数点で表す

浮動小数点表現では,たとえば 26.5 という数値は,  $0.265 \times 10^2$  表現できる.このとき, 0.265 を仮数部(mantissa, 小数点は左端にあると考え 26500...0),  $10^2$  を指数部(index, ただし, 16進数で表現)という(一般の浮動小数点表現  $a = f \times r^e$  においては,  $f$  を仮数,  $e$  を指数, そして  $r$  を基数などという.)

ここで,  $(26.5)_{10} = (11010.1)_2 = (1A.8)_{16}$  である. 小数点を左に 2 桁移動すると,  $1A.8 \rightarrow (0.1A8)_{16} \times 16^2$  となる. そこで, 基数が 16 であることの情報,  $(1000000)_2$  で表すとともに, 2 乗の情報を  $(10)_2$  を加えることによって, 結局,  $(1000010)_2$  で記す.

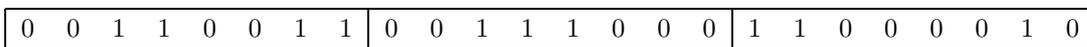
● ゾーン 10 進数

ゾーン 10 進数は, JIS コードや EBCDIC コードを用いて 1 バイトに 10 進数 1 桁を表現したものである. 1 バイトのうち上位 4 ビットをゾーンといい, データが数値であることを示すコード (JIS コードでは 0011) をおき, 下位 4 ビットに数値を入れる. 最下位桁のゾーン部には正, 負の符号を示す.

● パック 10 進数

パック 10 進数は, 10 進数 1 桁を半分の 4 ビットで表現し, 最下位の 4 ビットに正, 負の符号をつける. パック 10 進数は演算を行うときに用いられ, ゾーン 10 進数比べて効率のよい表現形式である.

382



-46

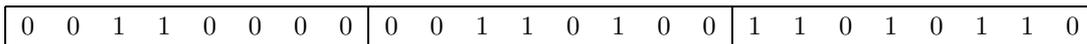


図 7: 382 と -46 を 3 バイトのゾーン 10 進数で表す

2735

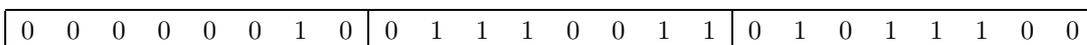


図 8: 2735 を 3 バイトのパック 10 進数で表す

## 2.5 画像や音声の表現

- マルチメディア (multimedia)

静止画や動画の画像，あるいは音声などのデータをデジタル化して，統一的に伝送・蓄積するようにした電気・磁気 / 電磁波等の媒体をマルチメディアという．最近では，このようなマルチメディアを PC などで再生・処理・加工ができるようになった．

- 音声 (voice, sound)

アナログ信号である音声をデジタル信号に変換 (Analog to Digital Converter, ADC, A/D 変換) して記録し，再生するときはその反対の変換 (Digital to Analog Converter, DAC, D/A 変換) を行って，アナログ信号である音声を取り出す．PCM (Pulse Code Modulation: パルス符号変調) である．

図9は， $f = 60 \text{ Hz}$  のアナログ信号 (正弦波:  $10 \sin 2\pi ft$ ) を 0.001 秒 (1ms) 周期 (サ

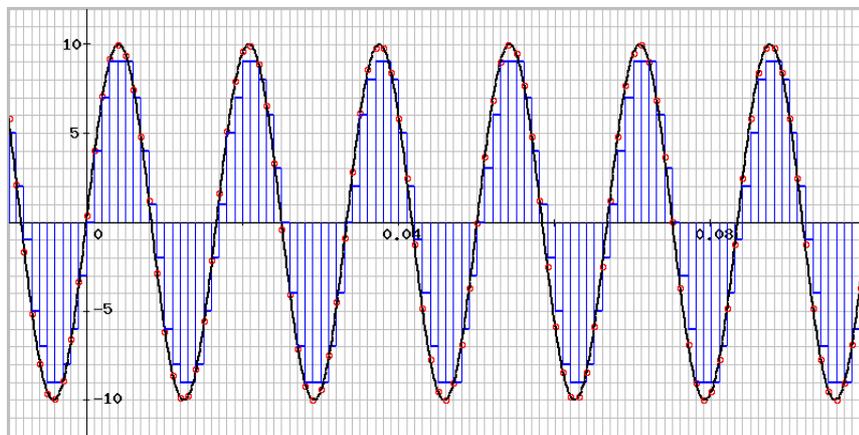


図9: A/D, D/A 変換

ンプリング周波数: 1 kHz) でサンプルし，量子化 (離散値化，整数化) してホールド (記憶保持) した場合の各波形を示した例である．時刻 0 からの量子化値は 10 進数で 0, 4, 7, 9, 9, 9, 7, 4, 1, -2, -6, -8, -9, ... となる．2 進数 (8ビット) では

0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	1	1	1
0	0	0	0	1	0	0	1
0	0	0	0	1	0	0	1
0	0	0	0	1	0	0	1
0	0	0	0	0	1	1	1
0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1
1	1	1	1	1	1	1	0
1	1	1	1	1	0	1	0
1	1	1	1	1	0	0	0
1	1	1	1	0	1	1	1

である．負数は 2 の補数で表している．

( 電話回線 (ISDN: Integrated Services Digital Network) は , 伝送周波数帯域 : 0.3 ~ 3.4 kHz , サンプル周波数 : 8 kHz (国際標準) , 符号化ビット数 : 8 bit , 伝送速度 : 64 kbit/s である .)

なお , 電子楽器とコンピュータの間で交換するための規格は , MIDI (Musical Instruments Digital Interface) とよばれる .

- 画像 (image)

#### 静止画

画像のフォーマットとしては EPS (Encapsulated PostScript Format) , すなわち , カプセルに入れられた PostScript が用いられる . さらに , それはドットとよばれる点で表現する方法 , ビットマップ (bitmap) に変換される . 写真等の静止画は , JPEG (Joint Photographic coding Experts Group) や GIF (Graphics Interchange Format) とよばれる方式によって圧縮して記憶し , 利用することが多い .

#### 動画

デジタルカメラ等によって撮影された動画は , 比較的短い時間に連続して静止画を表示し , 動いているように見せるものである . MPEG (Moving Picture coding Experts Group) や QuickTime とよばれる方式によって圧縮し , 利用することが多い .

図 10 は , ディスプレイ上において赤・緑・青の光の三原色の点で , 左肩より走査する (scan) する様子を表したものである .

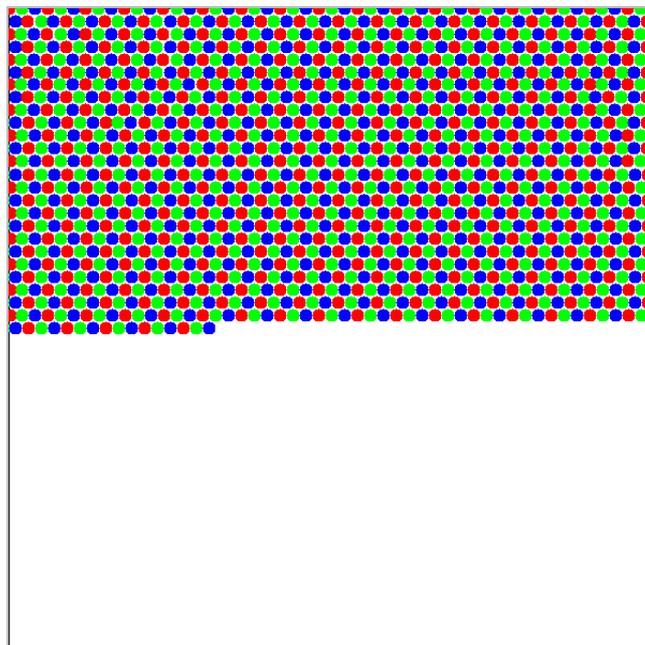


図 10: カラーパターン